

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Тараса Шевченка
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра програмних систем і технологій

Дисципліна
«Ймовірнісні основи програмної інженерії»

Лабораторна робота № 3

Виконав:	Шекера Олександр Валерійович	Перевірила:	Марцафей А.С.
Група	ІПЗ-22	Дата перевірки	
Форма навчання	денна	Оцінка	
Спеціальність	121		
2022			

Тема: Двовимірна статистика

Мета: Навчитись використовувати на практиці набуті знання про міри у двовимірній статистиці.

Завдання:

1. Намалюйте діаграму розсіювання для даних. Укажіть, чи існує тренд у даних. Якщо так, то вкажіть, чи є це негативним трендом, чи позитивним.
2. Знайдіть центр ваги і коваріацію.
3. Знайти рівняння лінії регресії у від х.
4. Розрахуйте коефіцієнт кореляції між даними.
5. Зробити висновок про залежності.

Математична модель:

Для побудови діаграми розсіювання та лінії регресії на графіку використовувалась бібліотека matplotlib мови програмування python.

Для визначення центру ваг використовувалась формула:

$$G(x, y) = G(\text{Mean}X, \text{Mean}Y)$$

Для визначення коваріації використовувалась формула:

$$C = \frac{\sum_{i=0}^n (x_i - \text{Mean}X) * (y_i - \text{Mean}Y)}{n}$$

Для визначення рівняння регресії використовувалась формула:

$$y = b_1x + b_0,$$

$$\text{Де } b_1 = \frac{C}{\text{Var}(x)},$$

$$b_0 = \text{Mean}Y - b_1 * \text{Mean}X$$

Для визначення коефіцієнта кореляції використовувалась формула:

$$Cor = \frac{\sum_{i=0}^n \left(\frac{x_i - MeanX}{s_x} * \frac{y_i - MeanY}{s_y} \right)}{n-1},$$

$$\text{Де } s_x = \sqrt{Var(x)},$$

$$s_y = \sqrt{Var(y)}$$

Код алгоритму:

```
import math
import matplotlib.pyplot as plt
import numpy as np

def task():
    fileName =
"C:\\Users\\oleks\\Desktop\\кн\\теорвер\\lab3\\task_03_data\\input_10.txt"
    f = open(f'{fileName}', 'r')
    length = int(f.readline())
    time = []
    cost = []

    for i in range(length):
        [item, item1] = f.readline().split()
        item = item.replace(',', '.')
        time.append(float(item))
        cost.append(int(item1))

    f.close()

    TotalTime = 0
    TotalCost = 0

    for i in range(length):
        TotalTime += time[i]
        TotalCost += cost[i]

    meanX = round((TotalTime / len(time)), 2)
    meanY = TotalCost / len(cost)

    covariance = 0
    varianceX = 0
    varianceY = 0

    for i in range(length):
```

```

        covariance += (time[i] - meanX) * (cost[i] - meanY)
        varianceX += pow((time[i] - meanX), 2)
        varianceY += pow((cost[i] - meanY), 2)
    covariance /= length
    varianceX /= length
    varianceY /= length

    b1 = covariance / varianceX
    b0 = meanY - b1 * meanX

    npX = np.array(time)
    npY = b1 * npX + b0

    standartDeviationX = math.sqrt(varianceX)
    standartDeviationY = math.sqrt(varianceY)
    correlation = 0

    for i in range(length):
        correlation += ((time[i] - meanX) / standartDeviationX) * ((cost[i] -
meanY) / standartDeviationY)

    correlation /= (length - 1)

    f =
open('C:\\Users\\oleks\\Desktop\\кнху\\теорвер\\lab3\\task_03_data\\output.txt',
'w')
    f.write(f'Center of gravity: ({meanX}, {meanY})\n')
    f.write(f'Covariance: {covariance}\n')
    f.write(f'Regression equation: y = {b1} + ({b0})*x\n')
    f.write(f'Correlation coefficient: {correlation}\n')

    if correlation > 0:
        f.write(f'Correlation is positive\n')
    elif correlation < 0:
        f.write(f'Correlation is negative\n')

    if correlation == 0:
        f.write('Conclusion: x and y are not linearly related\n')
    elif math.fabs(correlation) > math.sqrt(3) / 2:
        f.write('Conclusion: x and y have a strong linear relation\n')
    else:
        f.write('Conclusion: x and y have a weak linear relation\n')

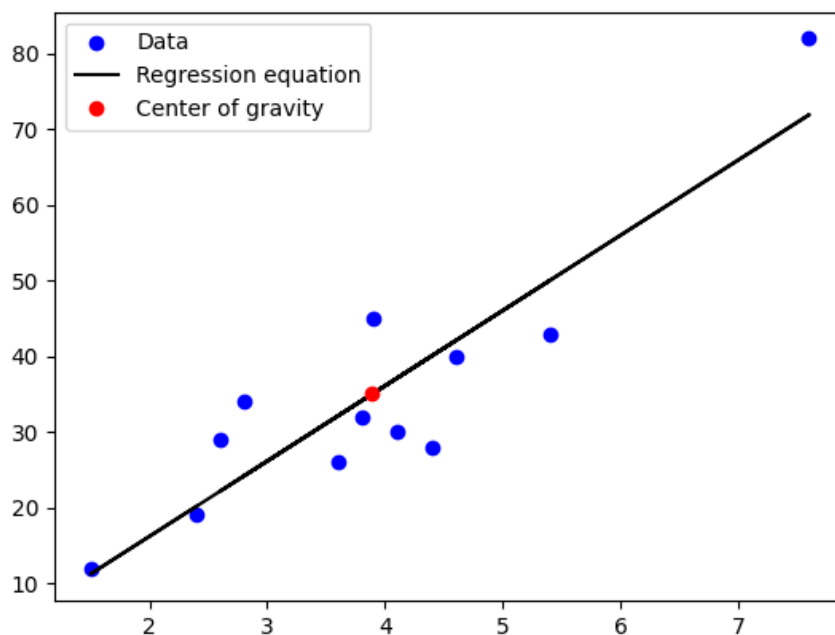
    plt.scatter(time, cost, c="blue", label="Data")
    plt.plot(npX, npY, c="k", label="Regression equation")
    plt.plot(meanX, meanY, 'ro', label="Center of gravity")
    plt.legend()
    plt.show()

task()

```

Випробування алгоритму:

Для перевірки запустимо програму на файлі, з 12 значеннями:



Center of gravity: (3.89, 35.0)

Covariance: 23.0

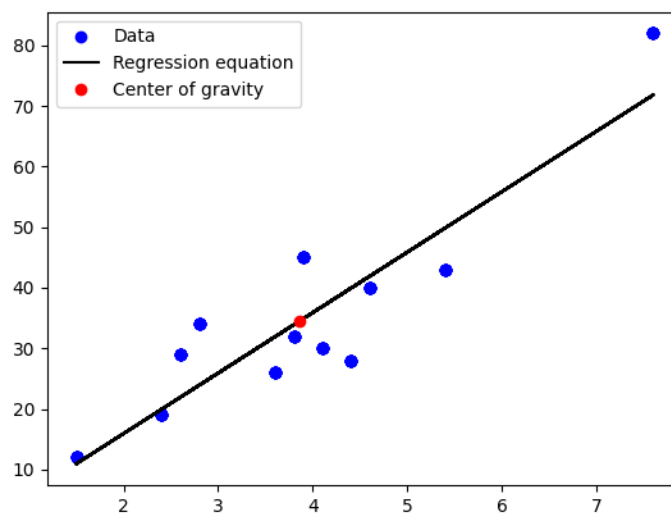
Regression equation: $y = 9.953406517317484 + (-3.718751352365011) \cdot x$

Correlation coefficient: 0.982910095376646

Correlation is positive

Conclusion: x and y have a strong linear relation

Результат роботи програми при вхідних даних розміром 100:



Center of gravity: (3.86, 34.5)
Covariance: 22.59200000000006
Regression equation: $y = 9.981973065639254 + (-4.030416033367523) * x$
Correlation coefficient: 0.9110577262104946
Correlation is positive
Conclusion: x and y have a strong linear relation

Висновок:

Виконано завдання третьої лабораторної роботи. Опановані навички використання на практиці набутих знань про міри у двовимірній статистиці. Розроблена програма, що малює діаграму розсіювання даних, знаходить центр ваг, коваріацію та рівняння лінії регресії, розраховує коефіцієнт кореляції між даними та робить висновок про залежності. Програма коректно працює на різних об'ємах вхідних даних.